

A Progressive Universal Noiseless Coder

Michelle Effros, *Student Member, IEEE*, Philip A. Chou, *Member, IEEE*,
Eve A. Riskin, *Member, IEEE*, and Robert M. Gray, *Fellow, IEEE*

Abstract—We combine pruned tree-structured vector quantization (pruned TSVQ) with Itoh's universal noiseless coder. By combining pruned TSVQ with universal noiseless coding, we benefit from the "successive approximation" capabilities of TSVQ, thereby allowing progressive transmission of images, while retaining the ability to noiselessly encode images of unknown statistics in a provably asymptotically optimal fashion. Noiseless compression results are comparable to Ziv-Lempel and arithmetic coding for both images and finely quantized Gaussian sources.

Index Terms—Progressive transmission, universal noiseless coding, medical image coding.

I. INTRODUCTION

UNIVERSAL noiseless coding has been investigated extensively in the literature [1], [2], as has progressive transmission of images [3]. In this paper, we present an integrated approach to both progressive and universal noiseless coding. We are motivated by the needs of the medical imaging community to store large amounts of image information losslessly and to access this information quickly at remote sites over slow communication links. The method of storage must not only be lossless in some applications, it must also be robust and efficient in terms of both space and accessibility. Universal noiseless coding provides robust and space-efficient storage without distortion, while progressive transmission, in which the decoder reconstructs increasingly better reproductions of the transmitted images, provides efficient access to remote sites over slow transmission links, and enables telebrowsing of large databases. Progressive transmission also provides the option of storing successive image refinements in slower, cost-effective media.

Pruned tree-structured vector quantization (pruned TSVQ) enjoys an optimal successive approximation property that makes it ideally suited for progressive transmission. This property allows a single tree structure to achieve minimal bit rates over a wide range of distortions. When the system is forced to zero distortion on finite precision images, however, achievable bit rates usually degrade sharply, often achieving expansion rather than the desired compression. This paper discusses the

combination of pruned TSVQ with a tree-structured universal noiseless code. This combination preserves the successive approximation nature of pruned TSVQ, while at zero distortion achieving provably asymptotically optimal compression on images of unknown statistics.

The tree-structured universal noiseless code with which pruned TSVQ is combined is due to Itoh [4]. His method is based on an approach to universal noiseless coding in which the encoder first sends to the decoder a model of the distribution of the data, and then describes the data using this model. The encoder chooses the model that minimizes the total description length of the data, i.e., the length of the model description plus the length of the data description given the model. Itoh uses a tree-structured model with leaf probabilities stored at each leaf to describe source statistics. By modifying Itoh's coder to get a simplified form of TSVQ, we combine the benefits of universal noiseless compression with the progressive transmission capabilities of pruned TSVQ.

In the next section, we review progressive image transmission techniques. In Section III, we discuss universal noiseless compression. We then combine progressive transmission and universal noiseless coding in Section IV. Both images and Gaussian sources are examined in Section V, and we find that noiseless compression rates approach the entropy rate regardless of their source (e.g., imaging modality).

II. PROGRESSIVE TRANSMISSION

In a progressive image transmission system, the decoder reconstructs increasingly better reproductions of the transmitted image as bits arrive over the channel. This allows for early recognition of the image, and has an obvious advantage in such applications as telebrowsing (scanning an image database): if the wrong image is being received, transmission can be aborted before the image is completely sent. This saves both bits and time.

Tzou presents a thorough review and comparison of a number of progressive transmission techniques [3]. To date, most work has involved progressive transmission of images compressed with lossy compression or uncompressed images. A notable exception is the work of Boncelet, who uses a quadtree-based approach for progressive transmission of losslessly compressed images [5]. A simple progressive transmission scheme with no overall compression is the bit-plane technique. In the bit-plane technique, the image is scanned and only the most significant bit of each pixel intensity is sent. The image is then repeatedly rescanned for remaining bits, each bit allowing for successively closer reproductions of the image, until the final image is lossless. The draft recommendations

Manuscript received November 10, 1992; revised May 18, 1993. This paper is based upon work supported in part by the National Science Foundation under an NSF Graduate Fellowship and NSF Grants MIP-9016974-A1 and MIP-9110508. This paper was presented in part at the IEEE International Symposium on Information Theory, Budapest, Hungary, June 1991.

M. Effros and R. M. Gray are with the Information Systems Laboratory, Stanford University, Stanford, CA 94305.

P. A. Chou is with the Xerox Palo Alto Research Center, Palo Alto, CA 94304.

E. A. Riskin is with the Department of Electrical Engineering, FT-10, University of Washington, Seattle, WA 98195.

IEEE Log Number 9215199.

of the Joint Bi-Level Image Experts Groups (JBIG) suggest binary compression of the bit planes as another means of progressive noiseless compression of both gray scale and color images [6]. Progressive transmission schemes that end in lossy compression include transform, subband, and pyramid techniques, and ordinary and pruned TSVQ. In the transform, subband, and pyramid techniques, the image is analyzed into frequency components by filtering and subsampling. The subsampled image is scanned, and only the lowest frequency components are sent. The image is repeatedly rescanned for the remaining frequency components. Since each component is quantized, the final image is lossy. More recently, pruned TSVQ [7], [8] has been used for progressive transmission. The image is scanned, and incremental descriptions corresponding to a sequence of nested, optimally pruned subtrees of a given TSVQ are sent to the decoder. TSVQ and pruned TSVQ both implement lossy compression, which we now describe.

In vector quantization, compression is achieved by mapping vectors or blocks of source data to the closest member (code-word) of a finite set (codebook) of reproduction vectors. In TSVQ [9], a tree is constructed with a reproduction vector at each node. During encoding, the distortions between a source vector and the reproduction vectors associated with each child of the root node are calculated. The encoder then descends to the “closest” child, and the process is repeated until the encoder reaches a terminal node. The encoder sends to the decoder a map of the path traveled, and then begins again with a new source vector. A path map may consist, for example, of a string of 1’s and 0’s describing left and right transitions in a binary tree, or the path map may be produced by an arithmetic code based on probabilities associated with the internal nodes. We will use the second of these methods. The decoder uses its own copy of the tree structure to trace the path to the desired terminal node, and then uses the reproduction vector associated with that node to represent the source vector.

A tree is generated for TSVQ using a training sequence typical of the source to be encoded. Starting with the root node, a node-splitting technique such as the generalized Lloyd algorithm [10], [11] is used to find a locally optimal (minimal distortion) set of m reproduction vectors by which the training vectors can be reproduced. These reproduction vectors are then placed at the m children of the root. The node-splitting technique is applied recursively to the children to produce a large m -ary tree (typically binary) with some maximum depth. Defining this initial tree as T and the set of terminal nodes associated with this structure as \tilde{T} , the codebook obtained using the full tree is the set of reproduction vectors stored at the nodes in \tilde{T} . Since reproduction vectors are also stored at internal nodes of the tree, intermediate reproductions of a single source vector can be constructed if the source is encoded first using smaller and later larger versions of the tree. By pruning a TSVQ [7], [12], it is possible to find the optimal nested set of subtrees to use in this progressive transmission process.

More specifically, let us call a tree S a *pruned subtree* of T and write $S \preceq T$ if the two trees share the same root and if S is a subset of T . Let $\mathbf{u}(S) = (r(S), d(S))$, where $r(S)$ and $d(S)$ are the tree functionals describing the average rate

and distortion associated with a pruned subtree S . Since any pruned subtree of T can itself be used to encode a data set, $\{\mathbf{u}(S) \mid S \preceq T\}$ is the set of points in the distortion-rate plane that can be achieved by T and its pruned subtrees. The operational distortion-rate function

$$\hat{D}_T(R) = \min_{S \preceq T} \{d(S) \mid r(S) \leq R\}$$

specifies the optimal tradeoff between rate and distortion in the set of T and its pruned subtrees. The function $\hat{D}_T(R)$ takes on a staircase form for which the convex hull can be found by minimizing the functional

$$J(S) = d(S) + \lambda r(S)$$

over $\{S \mid S \preceq T\}$. The multiplier λ is interpreted as the slope of the hyperplane supporting the achievable set of points $\mathbf{u}(S)$. The convex hull is a lower bound on the operational distortion-rate function, each of whose extrema is achieved by some pruned subtree S . Almost any point between extrema on the convex hull can be achieved by time sharing between the extremal subtrees. Let T_0 be the singleton tree consisting only of the root of T . Then $\mathbf{u}(T_0)$ will be the upper left corner of the convex hull since T_0 has the lowest average rate and highest average distortion of all pruned subtrees of T ; similarly, $\mathbf{u}(T)$ will be the lower right corner of the convex hull. The convex hull is traced by a nested sequence of pruned subtrees of T , each of which can be achieved from the previous by removing all nodes descending from a single internal node t .

The sequence of subtrees is efficiently derived by starting with T and pruning back to T_0 . Starting at any point $\mathbf{u}(S)$ on face F with slope λ of the convex hull, we must find the node at which we should prune to obtain the next subtree on the convex hull. Corresponding to each interior node $t \in S$, there is a pruned subtree $S(t)$ that would be obtained if all nodes of T below t were removed. Let $\Delta\mathbf{u}(S_t)$ equal the change in $\mathbf{u}(S)$ associated with removing from S all descendants of t . Then $\mathbf{u}(S) = \mathbf{u}(S(t)) + \Delta\mathbf{u}(S_t)$ for each $S(t)$. Each vector $\Delta\mathbf{u}(S_t)$ must have a slope $\Delta d(S_t)/\Delta r(S_t)$ no smaller in magnitude than λ since a slope with magnitude smaller than λ would correspond to a point below the convex hull. Further, as the convex hull is traced by a nested sequence of pruned subtrees, there exists a node t such that $\Delta\mathbf{u}(S_t)$ has exactly slope λ . The descendants of this node are removed, and then the process continues.

Each tree in the sequence is optimal in the sense that it codes the data to the lowest possible distortion among all the subtrees of T having the same or lower total description length. We therefore use this sequence to progressively code the data. More precisely, if $T_0 \prec T_1 \prec T_2 \prec \dots$ is the sequence of nested trees in order of increasing rate, then the overall encoding at stage $i+1$ is the overall encoding at stage i , followed by an encoding of those data whose path maps extend beyond T_i to T_{i+1} using an arithmetic code to encode the path map extensions. When the decoder receives the path map extension for a datum, it reconstructs the datum as the centroid of the appropriate leaf.

III. UNIVERSAL NOISELESS CODING

A universal noiseless code is a sequence of variable length codes whose expected lengths asymptotically achieve the entropy rate of the source for all sources in a class of sources [1]. More specifically, if $\{P_\theta: \theta \in \Lambda\}$ is a class of sources over the finite alphabet \mathcal{X} , and if $l(x^N)$ is the length function for the variable length noiseless code C_N on blocks of N letters from \mathcal{X} , then $\{C_N\}$ is a *weakly minimax* universal noiseless code if, for each $\theta \in \Lambda$, the per-letter redundancy

$$R_N(\theta) \triangleq \frac{1}{N} [E_\theta l(X^N) - H_\theta(X^N)]$$

goes to zero as $N \rightarrow \infty$, where H_θ is the entropy and E_θ is the expectation with respect to P_θ . $\{C_N\}$ is a *strongly minimax* universal noiseless code if the convergence is uniform in θ .

Davission has shown that for finite alphabet stationary ergodic sources, weakly minimax universal noiseless codes always exist [1]. He proves this by a construction due to Fitingof, in which the encoder blocks the sequence X^N into blocks of length L , sends to the decoder the histogram of these blocks, and finally uses the histogram to select a variable length noiseless code to transmit the sequence of blocks. For simplicity, we shall assume that the block length L evenly divides the data length N . The number of bits required to transmit the histogram is then $K^L \log(N/L + 1)$, where $K = |\mathcal{X}|$ is the alphabet size. This is because the histogram consists of K^L counts, each count taking a value in $0, \dots, N/L$. Given the histogram, the data can be coded to fewer than $H_\theta(X^L)$ bits per L -block, on average. (See the lemma in the Appendix.) Thus, the per-letter redundancy is at most

$$R_N(\theta) \leq K^L \log(N/L + 1)/N + H_\theta(X^L)/L - H_\theta(X^N)/N$$

where the difference between the second and third terms goes to zero as $N \rightarrow \infty$ provided $L \rightarrow \infty$ since P_θ is stationary and ergodic, and the first term goes to zero provided $L \rightarrow \infty$ at any rate slower than $\log(N/\log N)/\log K$.

If P_θ is independent and identically distributed (i.i.d.), then L need not go to infinity and $L = 1$ suffices. In this case, the per-letter redundancy is $K \log(N + 1)/N$. Indeed, Rissanen has shown that whenever P_θ is parameterized by K real numbers (i.e., $\Lambda \subseteq \mathcal{R}^K$), this is the fastest rate (within a factor of two) that the per-letter redundancy may approach zero for almost all $\theta \in \Lambda$ [2], [13]. In the i.i.d. case, these parameters are the letter probabilities p_1, \dots, p_K .

Now consider universal noiseless coding of a source of quantized real random variables. To be specific, suppose $\{\tilde{X}_i\}$ is a stationary ergodic source of real random variables on the unit interval $[0, 1]$ with process measure \tilde{P}_θ , $\theta \in \Lambda$, and suppose $X_i = q_K(\tilde{X}_i)$ is a quantized version of \tilde{X}_i , where q_K is a uniform scalar quantizer on $[0, 1]$ with K bins of width $\Delta = 1/K$. Then $\{X_i\}$ is a stationary ergodic source over a finite alphabet of size K with induced process measure P_θ , $\theta \in \Lambda$, and hence universal noiseless coding is possible, for example, with Fitingof's method of histogram encoding. If $\{\tilde{X}_i\}$ is i.i.d., then the per-letter redundancy of this method is $K \log(N + 1)/N$, as we have seen above. Thus, the redundancy increases without bound as the number of quantization levels K increases.

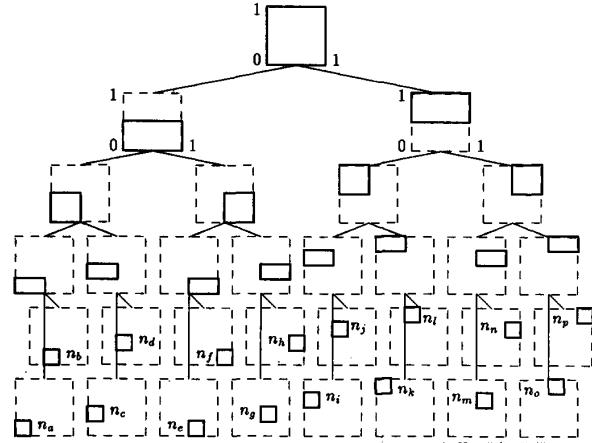


Fig. 1. Complete tree structure for $L = 2$ dimensions and $K = 4$ bins/dimension.

Itoh considers a method of universal noiseless coding of such quantized data in which the per-letter redundancy is bounded as the number of quantization levels increases [4]. In light of Rissanen's result, this can only be achieved with additional assumptions on \tilde{P}_θ . The additional assumption made by Itoh is that \tilde{P}_θ has a continuous, differentiable density with a finite maximum slope.

Itoh's method is similar to the histogram method in that the encoder blocks the sequence X^N into N/L blocks of length L , sends to the decoder a "histogram" of these blocks, and finally uses the "histogram" to select a variable length noiseless code to transmit the sequence of blocks. In Itoh's method, however, the "histogram" is actually a pruned tree-structured histogram. Consider the following. An ordinary histogram for this problem is the number of data blocks X^L falling into each of the K^L quantization bins defined by uniformly partitioning the L -dimensional unit cube $[0, 1]^L$ into K^L bins per dimension, each bin having volume $\Delta^L = (1/K)^L$. Assuming K is a power of two, it is possible to build a complete binary tree structure T on this histogram, as shown in Fig. 1 for $L = 2$ and $K = 4$, in which the leaves correspond to the quantization bins, the root corresponds to the unit cube, and each intermediate node corresponds to the union of all quantization bins for which the node is an ancestor. Thus, each node corresponds to a *cell*, or subset of the unit cube, and the cell of each node is partitioned by the cells of its children. The tree is arranged so that, starting with the root node, the cells are partitioned by the hyperplane perpendicular to a coordinate axis at the midpoint, taking the axes in turn. Thus, the tree is $L \log K$ levels deep, and all splits at level l are perpendicular to the $(l \bmod L)$ th axis. This tree may be pruned to obtain a pruned subtree S , as shown in Fig. 2, in which case the associated histogram is simply the number of data blocks X^L falling into each leaf cell. It is this pruned tree-structured histogram that Itoh transmits to the decoder.

The pruned tree-structured histogram is encoded by first transmitting the tree structure in preorder traversal format, using 1 b/node to specify whether the node is a leaf or not, and

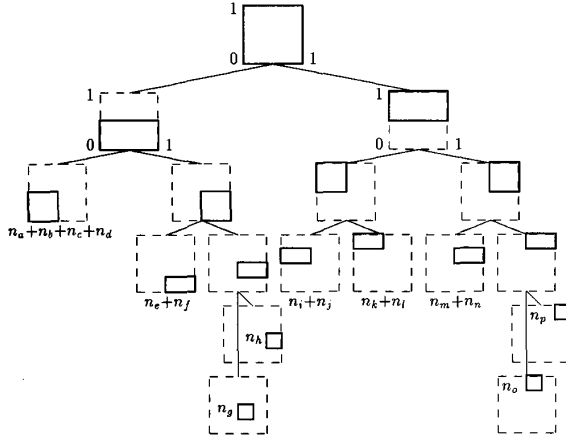


Fig. 2. Pruned tree structure for $L = 2$ dimensions and $K = 4$ bins/dimension.

then transmitting the $M - 1$ counts at the leaves, where M is the number of leaves. (The M th count can be determined from the others assuming a fixed image size.) Equivalently, $M - 1$ counts at the internal nodes could be transmitted instead of the leaf counts, where the internal node counts could represent the number of vectors going to, say, the left child. From these counts, all the leaf counts can be derived, and vice versa. In either case, each count requires the usual $\log(N/L + 1)$ bits, so the length of the description of the pruned tree-structured histogram S is

$$l(S) = \sum_{t \in S} 1 + \sum_{t \in S-\tilde{S}} \log(N/L + 1) \quad (1)$$

$$\begin{aligned} &= (2M - 1) + (M - 1) \log(N/L + 1) \\ &\leq M(\log(N/L + 1) + 2) \end{aligned} \quad (2)$$

bits. Itoh actually uses a more complicated scheme that codes S to half this length. (It turns out that the counts need not be transmitted to their full precision.) Asymptotically, however, this factor is not relevant, so we describe this simpler scheme.

Given the pruned tree-structured histogram S , each data block X^L is encoded in two parts. In the first part, the encoder follows the path in S to the leaf t in which X^L is contained and transmits the identity of the leaf using an arithmetic code matched to the leaf count. The number of bits required to specify the leaf is thus $-\log(n_t/n_0)$, where n_t is the number of L -blocks falling into leaf t , and $n_0 = N/L$ is the total number of L -blocks in X^N . [This is equivalent to transmitting the path map using an arithmetic code matched to counts n_i at each internal node along the path since $-\log(n_t/n_0) = -\sum_{i=1}^t \log(n_i/n_{i-1})$.] In the second part of the encoding, the encoder transmits the identity of the exact quantization bin within leaf t , using $\log(V_t/\Delta^L)$ bits, where V_t is the volume of leaf t and Δ^L is the volume of each quantization bin, $\Delta = 1/K$. The length of the description of the entire data sequence X^N given the pruned tree-structured histogram

S is thus

$$l(X^N | S) = \sum_{t \in \tilde{S}} n_t [-\log(n_t/n_0) + \log(V_t/\Delta^L)] \quad (3)$$

bits. This two-part encoding is nearly optimal if, within each leaf, the quantization bins are nearly equally likely. This will be true if the density of \tilde{X}^L across each leaf is nearly uniform.

The total description length, from (1) and (3), is therefore

$$l(X^N, S) \triangleq l(S) + l(X^N | S) = \sum_{t \in S-\tilde{S}} a_t + \sum_{t \in \tilde{S}} b_t \quad (4)$$

where

$$a_t = 1 + \log(N/L + 1),$$

$$b_t = 1 + n_t [-\log(n_t/n_0) + \log(V_t/\Delta^L)].$$

The pruned tree structure $T^* \preceq T$ that minimizes the total description length is the one used to encode X^N . Thus, the code length for X^N is the minimum description length

$$l(X^N) = l(X^N, T^*) = \min_{S \preceq T} l(X^N, S). \quad (5)$$

The minimum description length (5) and the tree that achieves it can be found efficiently as follows. First, extend $l(X^N, S)$ in (4) to all subtrees S of T (not necessarily pruned) with roots $t \in T$. Then $l(X^N, S)$ can be expressed recursively as

$$l(X^N, S) = a_t + \sum_{t' \in C(t)} l(X^N, S_{t'})$$

where the sum is over all children t' of the root t of S , and $S_{t'}$ is the subtree of S consisting of t' and all its descendants in S . The boundary condition for the recursion, when S consists only of its root node t , is

$$l(X^N, t) = b_t.$$

Thus, the minimum description length and the tree that achieves it can be found recursively by

$$\begin{aligned} l(X^N, T^*) &= \min_{S \preceq T} l(X^N, S) \\ &= \min \left\{ l(X^N, t), a_t + \sum_{t' \in C(t)} \min_{S \preceq T_{t'}} l(X^N, S) \right\} \\ &= \min \left\{ b_t, a_t + \sum_{t' \in C(t)} l(X^N, T_{t'}^*) \right\}. \end{aligned} \quad (6)$$

The computational complexity of this algorithm is linear in the number of nodes of the complete tree T .

It is intuitively clear that if the density of \tilde{X}^L is smooth, then as S gets larger, the density of X^L across each leaf approaches the uniform distribution, and the two-part encoding scheme for X^N given S becomes nearly optimal. Thus, there is a point of diminishing returns in which the cost of describing a larger S outweighs the improvement in coding X^N more efficiently. This turns out to imply that the optimal tree depth is bounded regardless of how finely X is quantized, and that consequently,

the overhead required to describe the tree, per-letter, goes to zero as the data length increases, even as the quantization is made finer and finer. This is made precise in the following.

Theorem: Let $\{\tilde{X}_i\}$ be a real-valued process on the interval $[0, 1]$ with stationary ergodic measure \tilde{P}_θ , $\theta \in \Lambda$, such that for each blocklength L , the marginal distribution \tilde{P}_θ^L has a continuous, differentiable density \tilde{p}_θ^L (with respect to Lebesgue measure) such that for all $x^L = (x_1, \dots, x_L)$, $\tilde{p}_\theta^L(x^L) \geq \epsilon_{L,\theta} > 0$ and $|\partial \tilde{p}_\theta^L(x^L) / \partial x_i| \leq A_{L,\theta} < \infty$. Let $\{X_i\}$ be the process $\{\tilde{X}_i\}$ uniformly scalar quantized to K bins per letter, with induced stationary ergodic measure P_θ . (K is a power of two.) Then Itoh's coder noiselessly codes sequences of length N from $\{X_i\}$ with per-letter redundancy no more than

$$R_N(\theta) \leq \left(\frac{\log(N/L + 1) + 2}{N} \right)^{1/2} + \frac{2LA_{L,\theta}^2}{\epsilon_{L,\theta}} \left(\frac{\log(N/L + 1) + 2}{N} \right)^{1/L} + H_\theta(X^L)/L - H_\theta(X^N)/N. \quad (7)$$

Proof: See the Appendix. A similar result has apparently been proved by Itoh in as yet unpublished work [14]. Our proof is included for convenience and completeness.

Corollary: If $A_{L,\theta}$ and $\epsilon_{L,\theta}$ do not, in fact, depend on $\theta \in \Lambda$, and $H_\theta(X^L)/L \rightarrow \bar{H}_\theta$ uniformly in K , where \bar{H}_θ is the entropy rate of P_θ , then Itoh's code is weakly minimax universal with the property that for each $\theta \in \Lambda$, the per-letter redundancy $R_N(\theta)$ goes to zero as $N \rightarrow \infty$ uniformly in the quantizer resolution K . If, furthermore, $H_\theta(X^L)/L \rightarrow \bar{H}_\theta$ uniformly in θ , then Itoh's code is strongly minimax universal with the property that the per-letter redundancy $R_N(\theta)$ goes to zero as $N \rightarrow \infty$ uniformly in both K and θ .

Proof: If $A_{L,\theta} = A_L$ and $\epsilon_{L,\theta} = \epsilon_L$, then there is a sequence $L_N \rightarrow \infty$ such that the first and second terms of (7) go to zero uniformly in both K and θ .

Remark: For fixed K , Itoh's coder was already clearly weakly minimax universal without conditions on P_θ other than stationarity and ergodicity (and it was already strongly minimax universal if $H_\theta(X^L)/L \rightarrow \bar{H}_\theta$ uniformly in θ) since its expected code length is at most a constant $2K^L - 1$ bits more than the expected code length for the histogram method, due to the one-bit-per-node description of the tree structure. Asymptotically, this constant is irrelevant.

Practically speaking, the import of the theorem and its corollary is that Itoh's code is a good universal code to use when the class of distributions P_θ , $\theta \in \Lambda$ is smooth relative to resolution of the quantizer. Thus, it is ideal in many signal compression applications.

IV. COMBINING PROGRESSIVE TRANSMISSION AND UNIVERSAL NOISELESS CODING

Pruned TSVQ and Itoh's universal noiseless coder combine naturally to produce a system capable of noiselessly encoding and progressively transmitting images with unknown source statistics. We will refer to the combined technique as progressive universal noiseless coding (PUNC).

PUNC uses an augmented version of Itoh's binary tree structure S to code the data X^N . In PUNC, the length $l(X^N | S)$ of the lossless description of X^N given S is identical to the description length of X^N given S in Itoh's coder (3). However, in PUNC, the description length $l'(S)$ of the tree S is slightly larger than the description length $l(S)$ in Itoh's coder (1) because of extra information used to specify reproduction vectors at each node and to transmit the tree itself progressively. Nevertheless, for all $S \preceq T$, $l'(S)$ is no more than a constant times $l(S)$; hence, the theorem in Section III holds with the first term in (7) multiplied by a constant, and the corollary holds as well. In particular, it will turn out that

$$l'(S) = L \log(N/L + 1) + \sum_{t \in S - \tilde{S}} [1 - \log(n_t/n_0)] + (L + 1) \log(N/L + 1) + \sum_{t \in \tilde{S}} [1 - \log(n_t/n_0)] \quad (8)$$

so that

$$\begin{aligned} l'(S) &\leq L \log(N/L + 1) \\ &\quad + \sum_{t \in S - \tilde{S}} [1 + (L + 2) \log(N/L + 1)] \\ &\quad + \sum_{t \in \tilde{S}} [1 + \log(N/L + 1)] \\ &\leq \sum_{t \in S - \tilde{S}} [1 + (L + 2) \log(N/L + 1)] \\ &\quad + \sum_{t \in \tilde{S}} [1 + (L + 2) \log(N/L + 1)] \\ &\leq (L + 2) \sum_{t \in S} [1 + \log(N/L + 1)] \\ &\leq 2(L + 2) \left[\sum_{t \in S} 1 + \sum_{t \in S - \tilde{S}} \log(N/L + 1) \right] \\ &= 2(L + 2)l(S). \end{aligned}$$

By summing (3) and (8), the total lossless description length becomes

$$\begin{aligned} l'(X^N, S) &= l'(S) + l(X^N | S) \\ &= L \log(N/L + 1) \\ &\quad + \sum_{t \in S - \tilde{S}} [1 - \log(n_t/n_0) + (L + 1) \log(N/L + 1)] \\ &\quad + \sum_{t \in \tilde{S}} [1 - \log(n_t/n_0) + n_t[-\log(n_t/n_0) + \log(V_t/\Delta^L)]] \\ &= L \log(N/L + 1) + \sum_{t \in S - \tilde{S}} a_t + \sum_{t \in \tilde{S}} b_t \\ &= L \log(N/L + 1) + l'_0(X^N, S) \end{aligned}$$

where now

$$a_t = 1 - \log(n_t/n_0) + (L + 1) \log(N/L + 1),$$

$$b_t = 1 - \log(n_t/n_0) + n_t[-\log(n_t/n_0) + \log(V_t/\Delta^L)],$$

and

$$\begin{aligned} l'_0(X^N, S) &= \sum_{t \in S - \tilde{S}} a_t + \sum_{t \in \tilde{S}} b_t \\ &= \begin{cases} b_t & \text{if } S = t \\ a_t + \sum_{t' \in C(t)} l'_0(X^N, S_{t'}) & \text{otherwise.} \end{cases} \end{aligned}$$

In this recursive formulation, $l'_0(X^N, S)$ has been extended to all subtrees S in T (not necessarily pruned) with roots $t \in T$.

Like Itoh's coder, PUNC selects the pruned tree structure $T^* \preceq T$ that minimizes the total description length:

$$\begin{aligned} l'(X^N, T^*) &= \min_{S \preceq T} l'(X^N, S) \\ &= L \log(N/L+1) + \min_{S \preceq T} l'_0(X^N, S) \\ &= L \log(N/L+1) \\ &\quad + \min \left\{ b_t, a_t + \sum_{t' \in C(t)} l'_0(X^N, T_{t'}) \right\}. \quad (9) \end{aligned}$$

As in (6), this can be efficiently performed in time $O(|T|)$.

Given the tree T^* that minimizes the total (lossless) description length, PUNC uses the optimal pruning algorithm of Section II to select a sequence of pruned subtrees, $T_0 \prec T_1 \prec T_2 \prec \dots \prec \bar{T}^*$, which can be used to represent X^N with greater and greater fidelity. Here, \bar{T}^* is the tree T^* extended by adding *one* child to each leaf of T^* . The leaves of \bar{T}^* (i.e., the newly added children) represent the L -blocks of X^N losslessly, while the interior nodes of \bar{T}^* (i.e., the nodes of T^*) represent the L -blocks of X^N with distortion using the reproduction vectors associated with those nodes. Each tree $T_i \preceq \bar{T}^*$ is optimal in the sense that it codes the data to the lowest possible distortion among all pruned subtrees of \bar{T}^* having the same or lower rate, or total description length. Notice that the tree-growing processes for TSVQ and PUNC differ: the tree used in TSVQ is grown by making a sequence of stepwise-optimal splits of the training data, while the tree used in PUNC is grown in a deterministic fashion. In neither case is the convex hull of the operational distortion rate function, which is traced during progressive transmission, guaranteed to converge to the true distortion rate function of the source.

Progressive transmission is accomplished as follows. First, the centroid of the entire data sequence is transmitted using $L \log(N/L+1)$ bits. This is the reproduction vector used at the root, and accounts for the first term in (8). With this initial tree T_0 , the decoder can already begin reconstructing the received image. In general, given tree T_i , tree T_{i+1} can be transmitted by transmitting the sequence of nodes that, when split, produce T_{i+1} from T_i . Specifying each such node t , which is in the interior of T_{i+1} and hence is in T^* , requires $-\log(n_t/n_0)$ bits using an entropy code matched to the node probability n_t/n_0 . (Recall that n_t is the number of L -blocks falling into node t and $n_0 = N/L$ is the total number of L -blocks in X^N .) In addition, each transmitted node $t \in T^*$ requires 1 b to specify whether the node is an internal node or leaf of T^* . These account for the $1 - \log(n_t/n_0)$ terms in (8). If t is an internal node of T^* , the decoder splits t into left and right children t_L and t_R , and if t is a leaf of T^* , the decoder extends t into a leaf

of \bar{T}^* . In the first case, an additional $(L+1) \log(N/L+1)$ bits are required to specify the count and the reproduction vector (centroid) of the left child t_L . This accounts for the remaining terms in (8). From this information, the decoder can derive the count and centroid of the right child t_R by $n_{t_R} = n_t - n_{t_L}$ and $v_{t_R} = (n_t v_t - n_{t_L} v_{t_L})/n_{t_R}$. Like the counts, the L coordinates of the reproduction vectors are quantized to only $\log(N/L+1)$ bits each. Itoh [4] and (more generally) Rissanen [2], [13] have shown that this is about twice the precision necessary for the counts; Chou *et al.* [15], [16] and Zeger *et al.* [17] have shown that this is about twice the precision necessary for the reproductions. However, we can be generous since this factor of two is asymptotically irrelevant. Finally, the description of T_{i+1} is followed by an encoding of those data whose path maps extend beyond T_i to T_{i+1} , using an arithmetic code to encode the path map extensions. When the decoder receives the path map extension for a vector, if the path map terminates at an interior node of \bar{T}^* , it reconstructs the datum as the centroid of the terminating node. However, if the path map terminates at a leaf of \bar{T}^* , the decoder reconstructs the datum noiselessly according to a fixed rate code (matched to the volume of the leaf) immediately following the path map. Each bit of this fixed rate code successively refines the reproduction value until the true value is described, thereby also allowing progressive transmission when the rate is between that of T^* and \bar{T}^* . Thus, progressive transmission is extended naturally from lossy to lossless compression.

The transmission procedure makes clear what tree functionals $d(S)$ and $r(S)$ should be used in the pruning algorithm. Let S be any pruned subtree of \bar{T}^* that might be used to represent a partial transmission of X^N . The distortion of this partial transmission is then

$$d(S) = \sum_{t \in \tilde{S}} D_t$$

where D_t is the total distortion in node t . This is zero if t happens to be a leaf of the augmented tree \bar{T}^* , but is otherwise the sum of the distortions between the centroid of t and input vectors falling into t . The length in bits of the partial transmission is

$$r(S) = l'(S^*) + l'(X^N | S)$$

where $l'(S^*)$ is the number of bits describing the tree and $l'(X^N | S)$ is the number of bits describing the data given the tree. Here, with $S^* = S \cap T^* \preceq T^*$, $l'(S^*)$ is defined in (8) and $l'(X^N | S)$ is defined as

$$\begin{aligned} l'(X^N | S) &= \sum_{t \in S^* - \tilde{S}^*} [-n_{t_L} \log(n_{t_L}/n_t) - n_{t_R} \log(n_{t_R}/n_t)] \\ &\quad + \sum_{t \in \tilde{S} \cap \bar{T}^*} n_t \log(V_t/\Delta^L) \\ &= \sum_{t \in \tilde{S}^*} -n_t \log(n_t/n_0) + \sum_{t \in \tilde{S} \cap \bar{T}^*} n_t \log(V_t/\Delta^L). \end{aligned}$$

For $S = \bar{T}^*$, $l'(X^N | \bar{T}^*) = l(X^N | T^*)$, where $l(X^N | T^*)$ is defined in (3), so that as claimed, the lossless description

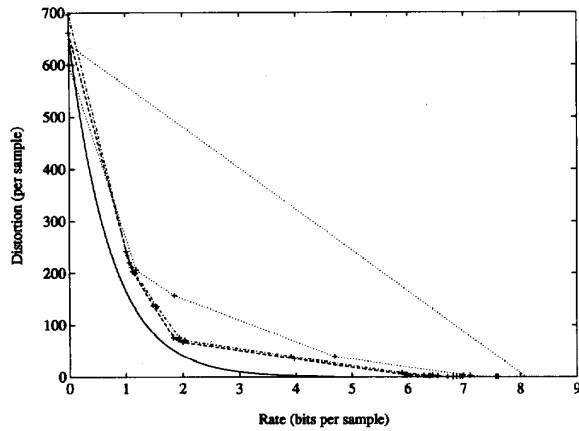


Fig. 3. Results on Gaussian data. Curves from top to bottom: results on 10, 100, 1000, and 10 000, samples and ideal distortion-rate curve.

TABLE I
PUNC RESULTS ON 8-B QUANTIZED GAUSSIAN
DATA WITH MEAN=128, STD. DEV. = 25.6

Noiseless Rates on Gaussian Data	
No. of Samples	Rate
10	7.873 bits/sample
100	6.958 bits/sample
1000	6.849 bits/sample
10000	6.789 bits/sample
Entropy	6.725 bits

length of X^N given S in PUNC is the same as the total description length of X^N given S in Itoh's coder.

V. RESULTS

The progressive universal noiseless coder described here inherits benefits from both its noisy and noiseless predecessors. Tests were performed on both synthetic Gaussian sources and images of several modalities.

The distortion-rate curves for progressive universal noiseless coding of various numbers of samples from a single one-dimensional independent Gaussian source (mean 128, standard deviation 25.6) that is uniformly scalar quantized to 8 bits are shown in Fig. 3, along with the ideal distortion-rate curve (solid line). As the number of samples N increases, the per-letter total description length goes to the entropy rate as summarized in Table I. Note also that PUNC traces the ideal distortion-rate curve with increasing precision as N increases, thereby illustrating the added benefit achieved with progressive transmission.

Experiments were also performed on three 8 b/pixel (bpp) gray scale images: an image from the USC database, a magnetic resonance brain scan, and a digitized mammogram. The sizes of both the USC image and the brain scan are

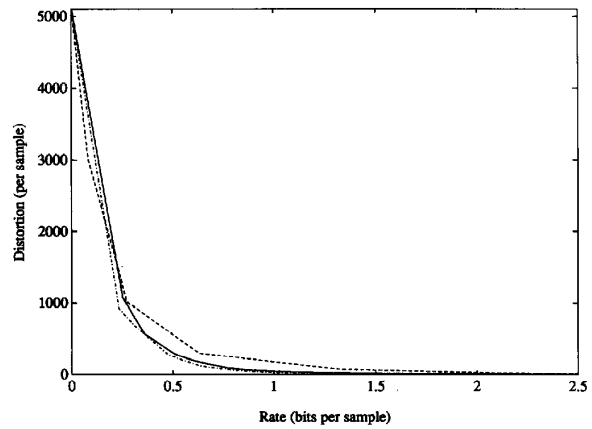


Fig. 4. PUNC, an arithmetic coding, and pruned TSVQ on mammogram. --- PUNC, — an arithmetic code, —. — pruned TSVQ.

TABLE II
PUNC RESULTS ON IMAGE DATA (ORIGINALLY 8 b/PIXEL)

Image	Ziv-Lempel	Arithmetic	PUNC
usc	6.7758 bpp	5.6266 bpp	6.5805 bpp
brain scan	7.0140 bpp	5.2819 bpp	6.3721 bpp
mammogram	6.6659 bpp	5.1708 bpp	5.7359 bpp

256×256 pixels, while the size of the mammogram is 1024×1024 pixels. Table II shows noiseless compression rates achieved by Ziv-Lempel coding (as implemented by the UNIX[®] utility compress), an arithmetic coding (using a simple first-order Markov context on the exclusive OR of each bit plane with the previous bit plane), and PUNC (with vector dimension 2×2 pixels). The noiseless compression rates are comparable, particularly for the larger medical image, but in all cases, arithmetic coding outperforms PUNC which outperforms Ziv-Lempel. Fig. 4 shows the distortion-rate curves achieved by PUNC, an arithmetic coding of the exclusive OR of the bit planes, and pruned TSVQ on the digitized mammogram. In the case of pruned TSVQ, the tree structure was grown and tested on the same image to provide a high standard against which we can compare intermediate image quality of the other two noiseless progressive transmission techniques. PUNC produces a distortion-rate performance superior to that of our simple arithmetic coding for a large fraction of the image reconstruction period, although this advantage is less apparent visually. PUNC also produces a low bit-rate performance close to that of pruned TSVQ, while simultaneously guaranteeing the eventual distortion-free reproduction of the original image. PUNC therefore presents a viable technique for combining the benefits of noiseless compression and progressive transmission.

VI. CONCLUSION

We have described a progressive universal noiseless coder. This coder combines Itoh's universal noiseless coder with

pruned TSVQ to provide noiseless data compression while allowing for progressive image transmission. We prove that the method inherits the universal property of Itoh's technique. Simulations on synthetic Gaussian sources support these conclusions by showing that the noiseless code length achieved by PUNC approaches the entropy of the uniformly quantized process. These simulations also demonstrate that the gap between distortion-rate curves measured with PUNC and the distortion-rate function narrows with increasing sample size. Tests on real images show that PUNC provides compression ratios comparable to those given by Ziv-Lempel and arithmetic coding. The quality of low rate intermediate images achieved by PUNC is close to what can be obtained with pruned TSVQ.

APPENDIX

Proof of Theorem

Let T be the complete binary tree-structured histogram of depth $L \log K$. We will construct a pruned tree-structured histogram \hat{T} of T such that its description length $l(\hat{T})$ is at most

$$l(\hat{T}) \leq \sqrt{N(\log(N/L + 1) + 2)} \quad (10)$$

and the expected description length $El(X^N | \hat{T})$ of the data given \hat{T} is at most

$$El(X^N | \hat{T}) \leq \frac{N}{L} \left[H_\theta(X^L) + \frac{2(LA_{L,\theta})^2}{\epsilon_{L,\theta}} \cdot \left(\frac{\log(N/L + 1) + 2}{N} \right)^{1/L} \right]. \quad (11)$$

Dividing the total expected description length by N proves the theorem since T^* is always at least as good as \hat{T} by the definition of T^* (5).

The pruned tree-structure \hat{T} that we will construct is actually a complete tree of depth Ld , where $d = \min(d', \log K)$ and d' is an integer satisfying

$$2^{-L(d'+1)} \leq \sqrt{(\log(N/L + 1) + 2)/N} < 2^{-Ld'}. \quad (12)$$

Thus, the number of leaves in \hat{T} is bounded by

$$M = 2^{Ld} \leq 2^{Ld'} < \sqrt{N/(\log(N/L + 1) + 2)}$$

and, using (2), the description length of \hat{T} is bounded by

$$l(\hat{T}) < \sqrt{N(\log(N/L + 1) + 2)}$$

which satisfies (10).

It remains to show that (11) is satisfied. First, we treat the case $d = \log K$, and then we treat the case $d < \log K$.

If $d = \log K$, then $\hat{T} = T$ is a complete tree-structured histogram for the L -blocks of X^N , and by the following lemma (with $Y = X^L$, $n = N/L$, and $M = K^L$), we have

$$El(X^N | \hat{T}) \leq (N/L)H_\theta(X^L)$$

which satisfies (11).

Lemma: Let Y^n be a sequence of n (not necessarily independent) identically distributed discrete random variables on M letters with probability mass function $p = (p_1, \dots, p_M)$, and let $l(Y^n | n_1, \dots, n_M) = -\sum_m n_m \log(n_m/n)$ be the length of the description of Y^n using an entropy code matched to the histogram n_1, \dots, n_M of Y^n . Then $El(Y^n | n_1, \dots, n_M) \leq nH(Y)$. (Note that in the expectation, the counts n_1, \dots, n_M are also random variables.)

Proof of Lemma: Let $l(Y^n | p) = -\sum_m n_m \log p_m$ be the length of the description of Y^n using an entropy code matched to the probabilities p_1, \dots, p_M . Then

$$\begin{aligned} l(Y^n | p) - l(Y^n | n_1, \dots, n_M) \\ = n \sum_m (n_m/n) \log \frac{(n_m/n)}{p_m} \geq 0. \end{aligned}$$

Taking expectations yields the lemma.

Now, assume $d < \log K$. Then $\hat{T} \prec T$ is a complete tree-structured histogram (of depth $Ld < L \log K$) for the L -blocks of \hat{X}^N , where \hat{X}^N is the sequence of real random variables \hat{X}^N uniformly scalar quantized to $d < \log K$ bits of precision. (Recall that X^N is the sequence of real random variables X^N uniformly scalar quantized to $\log K$ bits of precision.) Thus, \hat{X}_i is a strictly coarser quantization of X_i than X_i , and in fact, $\hat{X}_i = q_{2^d}(X_i)$.

Given \hat{T} , Itoh encodes X^N using

$$l(X^N | \hat{T}) = l(\hat{X}^N | \hat{T}) + (N/L)(L \log K - Ld)$$

bits. By the above lemma, however, we have seen that

$$El(\hat{X}^N | \hat{T}) \leq (N/L)H(\hat{X}^L).$$

Hence,

$$El(X^N | \hat{T}) \leq (N/L)[H(\hat{X}^L) + (L \log K - Ld)]. \quad (13)$$

The code length $(L \log K - Ld)$ is matched to a conditionally uniform distribution, which we now exhibit. For notational simplicity, let $X = X^L$ and $\hat{X} = \hat{X}^L$. Also, let $P(x) = P_\theta^L(x)$, and let $P(\hat{x})$ and $P(x | \hat{x})$ be the obvious probability and conditional probability measures induced by $P(x)$. The following probability measure Q agrees with P on \hat{x} , but is uniform on x given \hat{x} :

$$Q(x) = Q(x, \hat{x}) = Q(x | \hat{x})Q(\hat{x}) = 2^{-(L \log K - Ld)}P(\hat{x}).$$

Thus, the relative entropy of P with respect to Q is

$$\begin{aligned} D(P \| Q) &= \sum_x P(x) \log \frac{P(x)}{Q(x)} \\ &= \sum_x P(x) \log \frac{P(x | \hat{x})}{2^{-(L \log K - Ld)}} \\ &= (L \log K - Ld) - H(X^L | \hat{X}^L). \end{aligned} \quad (14)$$

Combining (14) with (13), we get

$$\begin{aligned} El(X^N | \hat{T}) &\leq (N/L)[H(\hat{X}^L) + H(X^L | \hat{X}^L) + D(P \| Q)] \\ &= (N/L)[H(X^L) + D(P \| Q)]. \end{aligned} \quad (15)$$

Now, consider the densities \bar{p} and \bar{q} with respect to Lebesgue measure on $[0, 1]^L$ that are piecewise constant across the K^L quantization bins B_x indexed by x , but respectively agree with the discrete measures P and Q on these bins. That is,

$$\bar{p}(\tilde{x}) = \sum_x 1_{B_x}(\tilde{x}) P(x) / V(B_x) \quad (16)$$

and

$$\bar{q}(\tilde{x}) = \sum_x 1_{B_x}(\tilde{x}) Q(x) / V(B_x) \quad (17)$$

where $V(B_x)$ is the volume of B_x . It is easy to see that

$$\begin{aligned} D(\bar{p} \parallel \bar{q}) &= \int \bar{p}(\tilde{x}) \log \frac{\bar{p}(\tilde{x})}{\bar{q}(\tilde{x})} d\tilde{x} \\ &= \sum_x \frac{P(x)}{V(B_x)} \log \frac{P(x)/V(B_x)}{Q(x)/V(B_x)} \cdot V(B_x) \\ &= \sum_x P(x) \log \frac{P(x)}{Q(x)} \\ &= D(P \parallel Q). \end{aligned}$$

Hence, (15) becomes

$$El(X^N \mid \hat{T}) \leq (N/L)[H(X^L) + D(\bar{p} \parallel \bar{q})]. \quad (18)$$

Next, consider the density \tilde{p} with respect to Lebesgue measure on $[0, 1]^L$ that is the continuous, differentiable density of \tilde{P}_θ^L with maximum slope $A = A_{L,\theta}$ and minimum bound $\epsilon = \epsilon_{L,\theta}$. Since

$$P(x) = \int_{B_x} \tilde{p}(\tilde{x}) d\tilde{x},$$

(16) can also be written

$$\bar{p}(\tilde{x}) = \sum_x 1_{B_x}(\tilde{x}) \int_{B_x} \tilde{p}(\tilde{x}') d\tilde{x}' / V(B_x).$$

The point is that \bar{p} is the average of \tilde{p} within the quantization bin B_x , so that within each B_x , $\min_{B_x} \tilde{p} \leq \bar{p} \leq \max_{B_x} \tilde{p}$. *A fortiori*, within each of the coarser bins $B_{\hat{x}}$ indexed by \hat{x} ,

$$\min_{B_{\hat{x}}} \tilde{p} \leq \bar{p} \leq \max_{B_{\hat{x}}} \tilde{p}. \quad (19)$$

Likewise, since

$$Q(\hat{x}) = P(\hat{x}) = \int_{B_{\hat{x}}} \tilde{p}(\tilde{x}) d\tilde{x},$$

(17) can also be written

$$\begin{aligned} \bar{q}(\tilde{x}) &= \sum_{\hat{x}} 1_{B_{\hat{x}}}(\tilde{x}) Q(\hat{x}) / V(B_{\hat{x}}) \\ &= \sum_{\hat{x}} 1_{B_{\hat{x}}}(\tilde{x}) \int_{B_{\hat{x}}} \tilde{p}(\tilde{x}') d\tilde{x}' / V(B_{\hat{x}}). \end{aligned}$$

Here, we have also used the fact that Q is uniform on x given \hat{x} , so that $Q(x)/V(B_x) = Q(\hat{x})/V(B_{\hat{x}})$. Thus, \bar{q} is the average of \tilde{p} within each coarse bin $B_{\hat{x}}$, so that within each $B_{\hat{x}}$,

$$\min_{B_{\hat{x}}} \tilde{p} \leq \bar{q} \leq \max_{B_{\hat{x}}} \tilde{p}. \quad (20)$$

Furthermore, each coarse bin $B_{\hat{x}}$ has side 2^{-d} , so that

$$\max_{B_{\hat{x}}} \tilde{p} - \min_{B_{\hat{x}}} \tilde{p} \leq LA2^{-d} \quad (21)$$

where A is the maximum slope of \tilde{p} . Combining (19), (20), and (21), we see that the difference $h = \bar{p} - \bar{q}$ has magnitude at most

$$|h(\tilde{x})| \leq LA2^{-d} \quad (22)$$

for all $\tilde{x} \in [0, 1]^L$.

Using Taylor's formula and the calculus of variations, we now show that

$$D(\bar{p} \parallel \bar{q}) \leq \frac{(LA2^{-d})^2}{2\epsilon}. \quad (23)$$

Let $f(p) = D(p \parallel q)$ where, for notational simplicity, $p = \bar{p}$ and $q = \bar{q}$. Then, by Taylor's formula with $p = q + h$,

$$f(q + h) = f(q) + f'(q)h + (1/2)h^t f''(q + \alpha h)h$$

for some $0 \leq \alpha \leq 1$, where we have used vector notation for the linear functional $f'(p)h$ and the quadratic form $h^t f''(p)h$. Consider the first term. Clearly,

$$f(q) = 0$$

since $D(q \parallel q) = 0$. Now, consider the second term. Since

$$\begin{aligned} f'_{\tilde{x}}(p) &= \frac{\partial f(p)}{\partial p(\tilde{x})} = \frac{\partial}{\partial p(\tilde{x})} \int p(\tilde{x}) \ln \frac{p(\tilde{x})}{q(\tilde{x})} d\tilde{x} \\ &= 1 + \ln \frac{p(\tilde{x})}{q(\tilde{x})}, \end{aligned}$$

we have $f'_{\tilde{x}}(q) = 1$ for all \tilde{x} , and

$$f'(q)h = \int 1 \cdot h(\tilde{x}) d\tilde{x} = \int p - \int q = 1 - 1 = 0.$$

Finally, consider the third term. Since

$$f''_{\tilde{x}\tilde{x}}(p) = \frac{\partial^2 f(p)}{\partial (p(\tilde{x}))^2} = \frac{1}{p(\tilde{x})},$$

we have $f''_{\tilde{x}\tilde{x}}(q + \alpha h) \leq 1/\epsilon$ for all \tilde{x} , and

$$\begin{aligned} h^t f''(q + \alpha h)h &= \int h^2(\tilde{x}) / (q(\tilde{x}) + \alpha h(\tilde{x})) d\tilde{x} \\ &\leq \int h^2(\tilde{x}) / \epsilon d\tilde{x} \leq (LA2^{-d})^2 / \epsilon. \end{aligned}$$

Here, we have used the facts that for any $\alpha \in [0, 1]$, $\min_{\tilde{x}} q + \alpha h \geq \min_{\tilde{x}} q + h = \min_{\tilde{x}} p \geq \min_{\tilde{x}} \tilde{p} \geq \epsilon$, and that $h^2 \leq (LA2^{-d})^2$, from (22). Thus, (23) follows. Note that we obtain the same results, without using the calculus of variations, by discretizing the problem, using the ordinary multivariate version of Taylor's formula, and taking the limit of increasingly fine discretizations. However, that approach would involve more notation and would need to invoke a limit theorem for relative entropies.

Combining (23) with (18), we obtain

$$El(X^N \mid \hat{T}) \leq (N/L)[H(X^L) + (LA)^2 2^{-2d} / (2\epsilon)]. \quad (24)$$

However, $d = d'$ since $d < \log K$, so from (12), we have

$$2^{-Ld} \leq 2^L \{(\log(N/L + 1) + 2)/N\}^{1/2}.$$

Thus, (24) becomes

$$El(X^N | \hat{T}) \leq (N/L)[H(X^L) + (2(LA)^2/\epsilon) \cdot \{(\log(N/L + 1) + 2)/N\}^{1/L}],$$

which satisfies (11).

This completes the proof.

ACKNOWLEDGMENT

The authors wish to thank Prof. S. Itoh for his helpful comments during the preparation of this paper.

REFERENCES

- [1] L. D. Davisson, "Universal noiseless coding," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 783–795, Nov. 1973.
- [2] J. Rissanen, "Universal coding, information, prediction, and estimation," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 629–636, July 1984.
- [3] K.-H. Tzou, "Progressive image transmission: A review and comparison of techniques," *Opt. Eng.*, vol. 26, pp. 581–589, July 1987.
- [4] S. Itoh, "A source model for universal quantization and coding," in *Proc. 10th Symp. Inform. Theory and Its Appl.*, Enoshima Island, Japan, Nov. 1987, pp. 611–616 (in Japanese).
- [5] C. G. Boncelet, Jr., "The MWQT image compression algorithm," in *Proc. 24th Annu. Conf. Inform. Sci. Syst.*, Princeton, NJ, Mar. 1990, pp. 243–246.
- [6] CCITT Draft Recommendation T.82, ISO/IEC Committee Draft 11544, "Coded representation of picture and audio information—Progressive bi-level image compression," WG9-S1R4.1, Sept. 1991.
- [7] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Optimal pruning with applications to tree structured source coding and modeling," *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 299–315, Mar. 1989.
- [8] E. A. Riskin, T. Lookabaugh, P. A. Chou, and R. M. Gray, "Variable rate vector quantization for medical image compression," *IEEE Trans. Med. Imaging*, vol. 9, pp. 290–298, Sept. 1990.
- [9] A. Buzo, A. H. Gray, Jr., R. M. Gray, and J. D. Markel, "Speech coding based upon vector quantization," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 562–574, Oct. 1980.
- [10] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 129–136, Mar. 1982. Previously an unpublished Bell Lab. Tech. Note, 1957.
- [11] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84–95, Jan. 1980.
- [12] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees* (The Wadsworth Statistics/Probability Series). Belmont, CA: Wadsworth, 1984.
- [13] J. Rissanen, "Stochastic complexity and modeling," *Ann. Statist.*, vol. 14, pp. 1080–1100, Sept. 1986.
- [14] S. Itoh, personal communication, 1992.
- [15] P. A. Chou and M. Effros, "Rate and distortion redundancies for source coding with respect to a fidelity criterion," presented at the IEEE Int. Symp. Inform. Theory, San Antonio, TX, Jan. 1993.
- [16] P. A. Chou, M. Effros, and R. M. Gray, "A vector quantization approach to universal noiseless coding and quantization," *IEEE Trans. Inform. Theory*, in preparation.
- [17] K. Zeger, A. Bist, and T. Linder, "Universal source coding with codebook transmission," *IEEE Trans. Commun.*, submitted July 1992.